Four Pillars of Service-Oriented Architecture

Grace A. Lewis and Dr. Dennis B. Smith
Carnegie Mellon University, Software Engineering Institute

Among current technologies, Service-Oriented Architecture (SOA) has the greatest potential for implementing the vision of migration to net-centric operations. While SOA has been successful in many cases, it has also been marked by a number of expensive failures. This article outlines four pillars to SOA success that include the following: developing an appropriate SOA strategy, implementing effective SOA governance, making sound technology assessments, and accounting for the fact that SOA requires a different mindset. As a result, the article proposes how a Department of Defense (DoD) organization can develop and implement an effective strategy for SOA implementation.

A cornerstone of DoD policy for future software and systems policy is the migration of systems to net-centric operations. The net-centric vision requires the leveraging of a highly flexible set of capabilities that can be composed quickly and flexibly into applications that take advantage of the interoperable aspects of the web and provide effective mission value. Among current technologies, SOA has the greatest potential for implementing this vision.

However, there is a great deal of confusion about what SOA is, whether it is real, and what it takes to implement a SOA-based system. This article provides a high-level introduction to SOA, and then outlines how a DoD organization can develop an effective strategy for implementing the vision.

Basic SOA Concepts

SOA has become an increasingly popular mechanism for achieving interoperability between systems. It is a way of designing systems composed of services that are invoked in a standard way. Common goals for the adoption of SOA are to eliminate redundancy, assemble new functionality from existing services, adapt systems to changing needs, and leverage legacy investments. An SOA-based system is composed of the following:

- **Services:** These are reusable components that represent business or mission tasks, such as customer lookup, weather, sensor placement, account lookup, or credit card validation. Services can be globally distributed across organizations and reconfigured to support new tasks or missions. They are reusable because they can be utilized by many business processes or mission threads. They usually provide coarse-grained functionality, such as customer lookup as opposed to finer-grained functionality such as customer address lookup.
- **Service consumers:** These are clients for the functionality provided by the services, such as end-user applications, systems, or even other services.
- **SOA infrastructure:** The infrastructure connects service consumers to services. It usually implements a loosely coupled, synchronous or asynchronous, message-based communication model, but other mechanisms are possible. The infrastructure often contains elements to support service discovery, security, and other operations. A common SOA infrastructure is an Enterprise Service Bus (ESB) to support Web Service environments. The Army's System of Systems Common Operating Environment and Defense Information Systems Agency's Net-Centric Enterprise Services are two examples of SOA infrastructures within DoD.

The benefits of SOA can be significant. However, SOA implementation is a complex engineering task and requires careful attention to engineering issues as well as to the four pillars for SOA success that are presented in this article.

Pillars for Successful SOA-Based Systems Development

It is common to view SOA-based systems development as a technical problem with a technical solution. However, successful SOA-based systems development requires attention to four pillars as illustrated in Figure 1:

- Alignment with mission and business goals.
- Instantiation of principles of SOA governance.
- Evaluation of relevant technologies for SOA implementation.
- Recognition that SOA requires a different mindset than traditional development.

Strategic Alignment

The first pillar, *Strategic Alignment*, focuses SOA decision-making on mission and business priorities rather than the availability of vendor products, or preferences of individuals down the chain of command. If the wrong strategy is selected, it can result in an expensive collection of random services that are never used. A successful SOA strategy includes the following:

- Evidence of fulfillment of critical business goals.
- Alignment with organizational enterprise architecture and current and future Information Technology (IT) infrastructure.
- Realistic choices of technologies and infrastructures.
- Realistic and gradual adoption strategy.
- Adequate SOA governance structure.
- Priorities for implementation.
- Reuse strategy across internal and external organizations.

These issues can be addressed through activities that provide a focus to the SOA implementation, the overall business plan, identification of high priority business processes, and disciplined SOA adoption.

Focus to SOA Implementation

The high-level mission and business goals need to dictate the focus of an SOA implementation. As an example, four different high-level goals can lead to four different SOA strategies:

- An SOA-based system to support a battlefield will have critical needs to ensure performance, availability, and security.
- Increasing information available to stakeholders will focus on intuitive portals and creation of services related to information that is important to stakeholders.
- Integrating new partners will focus on a flexible SOA infrastructure, a very well-described service repository, and clear guidelines for composition.
- Maximizing security may lead to a proprietary SOA infrastructure.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
SEP 2007		**00-00-2007 to 00-00-2007**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Four Pillars of Service-Oriented Architecture	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Carnegie Mellon University ,Software Engineering Institute (SEI),Pittsburgh,PA,15213**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**Same as Report (SAR)**	**4**	

Overall Business Plan

At a high level, there is recognition that SOA can provide agility, adaptability, legacy leverage, and integration with business partners. Current work has identified the business value of SOA for E-Commerce [1], E-Services, banking, and on-line services. In order to determine the amount of investment required and the projected payoff, an economic analysis needs to be planned at the beginning of an SOA implementation to identify the following:

- What constitutes a success within the context of a specific SOA implementation?
- How is return on investment measured?
- What are the resulting savings of SOA implementation (e.g. infrastructure consolidation, server and application virtualization, reuse of services, business agility)?

Identification of High Priority Business Processes

Any organization has many potential business process tasks that are candidates for services. Services are identified through a top-down analysis of business and mission processes, a bottom-up legacy system inventory, or a combination of the two. High-priority services are selected based on their relationship to critical goals. Traditional business process modeling, business process analysis, and business process reengineering techniques can help to model business processes and identify areas where services may be valuable. Although these methods will not model services, they suggest a starting point for setting priorities. Some of these approaches include the following:

- Enterprise architecture – analyzes business goals, what the business does, the type of information needed, and how the business uses IT to meet its goals.
- Business process analysis – models the business and its relationship to the external environment. This is an approach for identifying business processes that are candidates to become services.
- Business process modeling – analyzes and optimizes business processes to optimize current performance. This can provide details on the modeling of specific processes once they have been identified as candidates.
- Business process reengineering – analyzes current business processes and changes these processes, often in a radical way, to meet new business needs.

Disciplined SOA Adoption

An SOA implementation can start with a *big bang* approach that attempts to get SOA implemented at once throughout an enterprise. However, it is more prudent to begin with a pilot project that will provide a proof of concept. Pilot projects should focus on areas that provide high impact and visibility with the lowest risk. Gradual implementation can then lead to other projects that integrate a single organizational unit, to projects that integrate multiple business units, and later to large scale efforts that provide a virtual enterprise where all applications are built based on services [2].

SOA Governance

Governance has been rated as the main inhibitor of SOA adoption [3]. SOA governance provides a set of policies, rules, and enforcement mechanisms for developing, using, and evolving SOA-based systems, and for analysis of their business value. SOA governance includes policies, procedures, roles, and responsibilities for design-time governance and runtime governance.

Design-time governance includes elements such as rules for strategic identification of services, development, and deployment of services, reuse, and legacy system migration to services. It also enforces consistency in use of standards, SOA infrastructure, and processes.

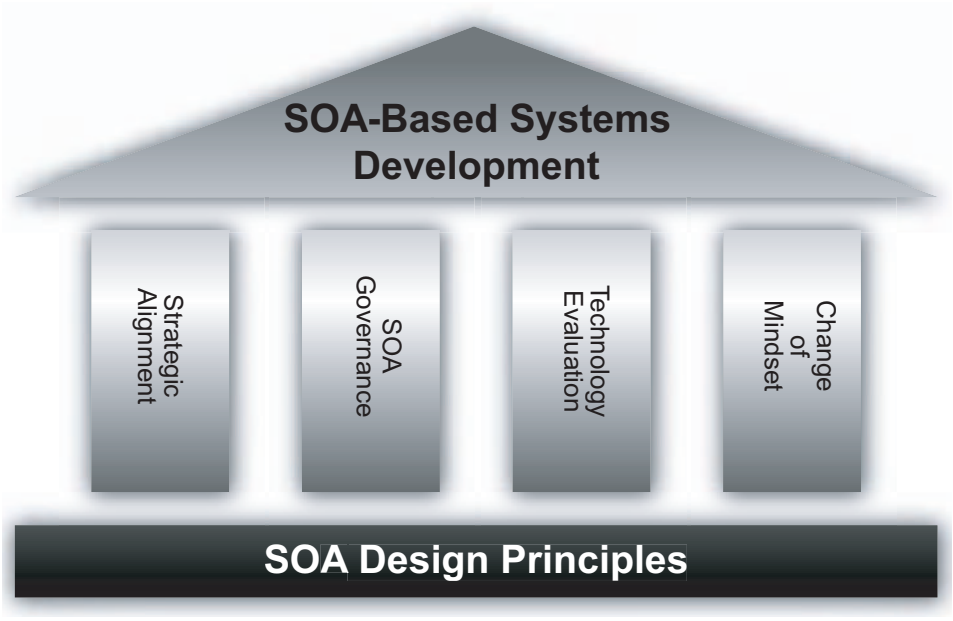
Runtime governance develops and enforces rules to ensure that services are executed only in ways that are legal. Runtime governance procedures address concerns such as 1) access to applications and data, 2) the replacement of services, and 3) consistent interactions with the SOA infrastructure.

Service-level agreements (SLAs) also fall under runtime governance. SLAs can include runtime validation of contractual specifications on performance, throughput, and availability; the use of automated metrics for tracking and reporting; and problem management.

A well-defined governance model needs to answer such questions as the following:

- What is the process for determining what services to create?
- What is the process for evolving and changing services if there are many consumers of the service?
- Many services can be common across several lines of business in an enterprise. Who *owns* these common services?
- Who owns the actual data if more than one service is using it?
- What is the resolution mechanism if there are conflicting requirements or change requests for shared services?
- What happens if the same (or similar) service is being developed by more than one service provider?
- What mechanisms, tools and policies are used for maintaining and monitoring deployed services?
- How are enterprise-wide policies enforced across various services both internally as well as externally to the organization?
- Who owns and maintains the shared repository of services in an organization?
- How are SLAs defined and enforced between service consumers and providers?

Figure 1: *Pillars of SOA-Based Systems Development*



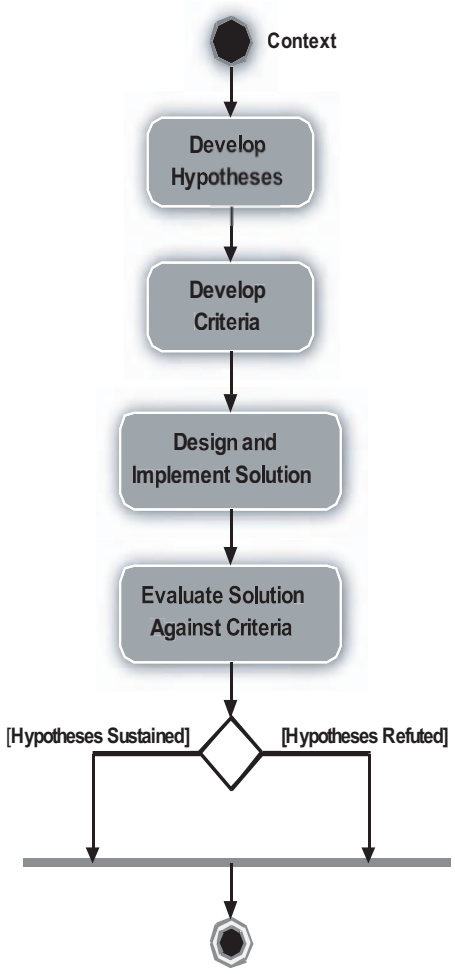


Figure 2: *T-Checks Approach for Technology Evaluation*

Technology Evaluation

Because an SOA implementation may use a number of technologies in novel contexts, it is important to evaluate whether a specific set of technologies is appropriate for the task at hand. Pillar 3 requires determining the fitness of a technology within a specific context before making a long term commitment to it. In adopting an SOA-based systems approach, a number of different technologies, standards and tools may be part of an implementation. Examples of these different technologies can involve specific web service standards, versions and tool implementations, custom infrastructures, ESBs, interfaces to specific databases, and language bindings.

It is easy to draw a slide showing how the pieces can fit together at an abstract level. However, all technologies work well within a specific context and under certain conditions. For example, Web services work well for asynchronous communication over the Internet. In a business environment these conditions are very common, but in a military tactical command and control environment this might not be the case because of high performance and availability requirements.

One way to perform this type of analysis is through a light weight evaluation method such as T-Checks [4, 5]. Other approaches can be used; however, the approach should enable a hands-on contextual analysis. The T-Checks approach is illustrated in Figure 2 and can be summarized in terms of the following steps:

- Identify technology requirements and context. Determine and document why the organization wishes to conduct the evaluation, what the expectations and concerns are with respect to the technology capabilities, and what is the context in which the technology plans on being used. Determine the environment in which the evaluation will take place, including expectations and constraints of the technology and measures of success.
- Develop hypotheses that are derived from the expectations placed on the technology. Hypotheses are claims about the technologies that are to be sustained or refuted.
- Develop criteria to determine if the results sustain or refute a hypothesis. Criteria are stated as a clearly measurable statement of capability. Each hypothesis can have one or more criteria, depending on the breadth covered by the hypothesis.
- Design and implement the experimental solution which is the simplest set of applications and/or components that are able to answer the questions posed by the hypotheses and associated criteria, within a given scenario. The experimental solution is implemented, run, and observed, until there is enough information to sustain or refute the set of hypotheses.
- Evaluate the solution against criteria in order to make a decision with respect to the fitness of the technology for the context in which it is intended to be used. Based on the results of the evaluation there should be enough information to decide if it is the following:
  - A good fit with requirements.
  - Not a good fit with requirements.
  - Has some mismatches that could potentially be solved by modifying the context or modifying the technology itself.

Awareness of a Different Mindset

There are a unique set of challenges in building SOA-based systems. These challenges require a different development approach that deals with the characteristics of SOA-based systems. Although it is difficult to generalize, some of the contrasts of SOA systems versus traditional systems are presented in Table 1.

These differences impact the way software is developed throughout the life cycle:

- During requirements, it is important to have close ties to business process modeling and analysis. In addition, there is the need to anticipate potential service requirements from unknown consumers.
- During architecture and design, it is important to have technology evaluations and to perform explicit trade-off analyses.
- Implementation decisions will be impacted by emerging standards and may require simulation of the deployment environment.
- Testing requires a strong emphasis on exception handling and requires all test instances of services are available.
- Maintenance requires more sophisticated impact analyses and greater coordination of release cycles.

Because SOA implementation requires a different mindset than traditional software development and acquisition, it is important to develop an overall transition strategy to address how to acquire the new skills that may be required through training personnel, hiring new staff, or bringing in external experts. In addition SOA merges the technical and business worlds; therefore, it is important to have expertise

Table 1: *Some Differences Between Traditional Systems Development and SOA-Based Systems Development*

Traditional Systems Development	SOA-Based Systems Development
Tight coupling between system components	Loose coupling between applications and services
Shared semantics at design time	Semantics ideally enable dynamic discovery and execution of services
Known set of users and usage patterns	Potentially unknown service users and usage patterns
System components all within the same organization	Multiple organizations providing and supporting system components

in both fields. The fact that SOA implementations have the potential of crossing the enterprise also suggests the need for developing a perspective that spans the concerns of the entire enterprise, rather than just the issues of a specific program or business unit. As discussed in the section Disciplined SOA Adoption, a gradual adoption process that starts with small scale pilots and expands gradually is also recommended.

Conclusions

The SOA approach offers real value for DoD organizations as a technology for migrating toward net-centric operations. However, the rhetoric surrounding SOA can often be confusing and misleading. Establishing an effective SOA approach is a complex acquisition, management and technical task. It requires the following:

- Alignment with mission and business goals.
- Instantiation of principles of SOA governance.
- Evaluation of relevant technologies for SOA implementation.
- Recognition that SOA requires a different mindset than traditional development.◆

References

1. Tilley, S., et al. "On the Business Value and Technical Challenges of Adopting Web Services." Journal of Software Maintenance and Evolution 16 (2004): 16, 31-50.
2. Schulte, R. "Meeting the Challenges of SOA Adoption." SOA in Action Virtual Conference, Nov. 2006.
3. "SOA Trend Survey." InfoWorld 2006.
4. Lewis, G., and L. Wrage. "A Process for Context-Based Technology Evaluation." Technical Note CMU/SEI-2005-TN-025. CMU/SEI, 2005.
5. Lewis, G., and L. Wrage. "Model Problems in Technologies for Interoperability: Web Services." Technical Note CMU/SEI-2006-TN-021. CMU/SEI, 2006.

About the Authors

Grace A. Lewis is a senior member of the technical staff at SEI. She is currently the lead for the System of Systems Engineering team within the Intermediate Systems to Intermediate Systems initiative. Lewis' current interests and projects are in SOA, legacy system modernization, and software development life-cycle activities in systems of systems. She has a bachelor's degree in systems engineering and an executive masters of business administration from Icesi University in Cali, Colombia, as well as a master's degree in software engineering from CMU.

SEI
4500 Fifth AVE
Pittsburgh, PA 15213
Phone: (412) 268-5851
Fax: (412) 268-5758
E-mail: glewis@sei.cmu.edu

Dennis B. Smith, Ph.D., is a senior member of the technical staff and is lead of the Integration of Software-Intensive Systems initiative at the SEI. This initiative focuses on developing and applying methods, tools, and technologies that enhance the effectiveness of complex networked systems and systems of systems. Smith has been involved with working with DoD organizations in developing an SOA capability, including issues of SOA strategy, governance and migration of legacy assets to SOA. He was the co-editor of the Institute of Electrical and Electronics Engineers and International Organization for Standardization-recommended practice on Computer-Aided Software Engineering Adoption, and has been general chair of two international conferences. Smith holds a masters degree and doctorate from Princeton University, and a bachelor's degree from Columbia University.

SEI
4500 Fifth AVE
Pittsburgh, PA 15213
Phone: (412) 268-6850
Fax: (412) 268-5758
E-mail: dbs@sei.cmu.edu